ATTACHMENT FOR CLAIM AMENDMENTS

The following is a marked up version of each amended claim in which underlines indicates insertions and brackets indicate deletions.

1 3. (Amended) A method according to claim 1 [or 2], wherein the step of creating new dimensional values and modifying the mapping comprises the steps of

examine whether the dimension is covering, and in case it is,

executing a make-onto procedure for making the dimension onto, thereby at least partly making an non-onto dimension aggregation normalised.

2 4. (Amended) A method according to claim 1 [any of claims 1-3], wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the dimension covering, thereby at least partly making a non-covering dimension aggregation normalised.

3 5. (Amended) A method according to claim 1 [any of claims 2-4], wherein the make-strict procedure comprises the steps of, starting from the bottom category and successively proceeding towards the top category,

identifying combinations of dimensional values of the same category for each of which combination at least one dimension value of a category below said category is linked to each of the dimension values of the combination,

creating one or more new dimensional values each representing one of the identified combinations of dimensional values and creating links from the new

dimensional values to dimensional values of above categories in accordance with existing links from each of the dimensional values represented by the new dimensional value, and

identifying dimension values being linked to identified combinations of dimensional values of the same category and replacing the links with links to new dimensional values representing said combinations of dimensional values.

4. 6. (Amended) A method according to claim 2 [any of claims 2-5], wherein the make-strict procedure comprises the successive steps of

(i)   setting the bottom category of the dimension as the child category,

(ii)   for each category being a direct predecessor of the child category of which category at least one dimension value of the child category is linked to a dimension value of, setting said category as the parent category and performing the steps of:

(iia)   ending the make-strict procedure for the parent category in case the parent category is the top category of the dimension,

(iib)   ending the make-strict procedure for the parent category in case no dimension value of the parent category is linked to a dimension value of a higher category,

(iic)   creating a new fused category in the dimension immediately below the parent category in case at least one of the dimension values of the child category is linked to more than one dimension value of the parent category,

(iid)   for each dimensional value of the child category, performing the steps of: creating a new dimension value of the new fused category representing the one or more

values of the parent category to which the dimensional value of the child category is linked and creating links from said new dimension value to said values in the parent category, the creation of the new dimension value being conditioned that no dimension value of the new fused category already exists having exactly such link(s), and

for each category being a direct predecessor of the parent category of which category at least one dimension value of the parent category is linked to a dimension value of, setting said category as a grandparent category and creating links from the new dimension value to the one or more dimension values of the grandparent category to which said one or more dimensional values of the parent category are linked,

(iie) removing the links from the parent category to the one or more grandparent categories, whereby the grandparent categories no longer are direct predecessors of the parent category,

(iif) creating links from each dimensional value of the child category to the dimension value of the new fused category having the same links to the dimension values of the parent category whereby the new fused category becomes a direct predecessor of the child category, and removing the links from the dimension values of the child category to the parent category, whereby the parent category no longer is a direct predecessor of the child category,

and

(iig) setting the new fused category as the child category and returning to step (ii).

5 ~7. (Amended) A method according to <u>claim 3</u> [any of claims 3-6], wherein the make-onto procedure comprises the steps of, starting from the top category and successively proceeding towards the bottom category,

creating, for each dimension value of each category above the bottom category not being linked to any dimensional value of the category immediately below, a new dimension value in the category immediately below and creating a link between said new dimension value and said dimension value of the category in question.

6 ~8. (Amended) A method according to <u>claim 3</u> [any of claims 3-7], wherein the make-onto procedure comprises the successive steps of

(i)   setting the top category of the dimension as the parent category,

(ii)   for each category immediately below the parent category and having dimension values being linked to dimension values of the parent category, setting said category as the child category and perform the steps of

(iia)   creating, for each dimension value of the parent category not being linked to any dimensional value of the child category, a new dimension value in the child category and creating a link between said new dimension value and said dimension value of the parent category,

(iib)   setting the child category as parent category,

(iic)   ending the make-onto procedure in case the parent category is the bottom category of the dimension, else returning to step (ii) of the make-onto procedure.

7 ~~9.~~ (Amended) A method according to <u>claim 4</u> [any of claims 4-8], wherein the make-covering procedure comprises the successive steps of

identifying links between dimension values of two categories having at least one intermediate category there between,

creating a new dimension value in each of said intermediate categories for each of those links for which no paths of links exists going only through immediate child-parent links from lower to higher categories and including a link to a dimension value of the intermediate category, and

replacing those links with links between the dimension values of those links and the new dimension values.

8 ~~10.~~ (Amended) A method according to <u>claim 4</u> [any of claims 4-9], wherein the make-covering procedure comprises the successive steps of

(i)   setting the bottom category of the dimension as the child category,

(ii)   for each category immediately above the child category for which at least one link between a dimension value of said category and a dimension value of the child category exists, setting the category as the parent category and perform the steps of:

(iia)   ending the make-covering procedure for the parent category in case the parent category is the top category of the dimension:,

(iib)   for each higher category being a direct predecessor category of the child category and being higher in the hierarchy than the parent category, performing the steps of

(iiba) identifying sets of dimension values of the higher category and

dimension values of the child category for which sets

a link exists, and

no paths of links going only from lower to higher categories and including

a link to a dimension value of the parent category exists, and

(iibb) creating for each identified set of dimension values a new dimension

value in the parent category, creating links between each of the dimension

values of the set and the new dimension value, and removing the link between

the two dimension values of the identified set, whereby the higher category no

longer is a predecessor of the child category,

(iic)   setting the parent category as the child category and returning to step (ii).

9. 11. (Amended) A method for by means of a computer to at least partly

aggregation normalise a multidimensional object including a set of facts comprising a

plurality of facts mapped on a plurality of dimensions having dimension values

organised into categories of dimension values based on a partial ordering, the

multidimensional object comprising mappings of links between dimension values within

each dimension, by means of applying the method of claim 1 [any of claims 1-10] to at

least one of the dimensions of the multidimensional object.

10. 13. (Amended) A method according to claim 11 [or 12], comprising the steps

of

selecting a subset of categories of the one or more dimension to be aggregation normalised, and

performing an aggregation normalisation of the selected subset,

whereby one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised.

14. (Amended) A method according to claim 11 [any of claims 11-13], comprising the steps of

selecting specific aggregation functions to be performed on the multidimensional object, and

selecting by means of the computer normalisation steps to be performed based on the selection of specific aggregation functions to be performed,

whereby one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised. .

17. (Amended) A method according to claim 15 [or 16], wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the multidimensional object covering, thereby at least partly making the non-covering multidimensional object aggregation normalised.,

13 18. (Amended) A method according to claim 15 [any of claims 15-17], wherein the method comprises the initial step of making each of the plurality of dimensions aggregation normalized [by means of the method according to any of claims 1-10].

14 19. (Amended) A method according to claim 11 [any of claims 11-18], wherein the created new dimensional values are marked as such, a pre-aggregation is performed on a multidimensional object being normalised by means of the computer and the method further comprises the step of

producing a reply to a query made to the system and concerning the multidimensional object, aggregate queries, exploring the dimension hierarchies, as well as navigation queries, that summarise the data at various levels of detail, in which reply the existence of the created new dimensional values is transparent.

15 20. (Amended) A method according to claim 11 [any of claims 11-19], further comprising the steps of

implementing, into the aggregation normalised multidimensional object, of new facts including mapping of the facts onto the dimension, of new dimension values of the dimensions, or of new mapping between some of the dimension values, by which implementation irregularities of the multidimensional object is introduced,

analysing the introduced irregularities of the dimensions of the multidimensional object,

creating new dimensional values of the multidimensional object and modifying

the mapping between dimensional values of the multidimensional object according to

the analysis, whereby the multidimensional object is aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of

the computer.

16 21. (Amended) A computer system comprising at least one general purpose

computer having data storage means associated therewith on which data storage

means is stored a computer programme product suitable for adapting the computer to

perform an at least partly aggregation normalisation of a multidimensional object

according to the method of claim 11 [any of claims 11-20], the computer system

comprising means for retrieving the computer programme product and perform

accordingly.

17 22. (Amended) A computer programme product suitable for adapting a

general purpose computer to perform an at least partly aggregation normalisation of a

multidimensional object according to the method of claim 11 [any of claims 11-20].

18 25. (Amended) A computer system according to claim 23 [or 24], wherein the

query handler component is adapted for producing replies to aggregate queries in which

replies the existence of the second plurality of dimensions is transparent.

19   28.   (Amended) A computer system according to <u>claim 23</u> [any of claims 23-27] further comprising means adapted for performing an at least partly aggregation normalisation of a multidimensional object [according to the method of any of claims 11-20].